AMRAD Newsletter

AMATEUR RADIO RESEARCH AND DEVELOPMENT CORPORATION

P.O. Drawer 6148 *McLean, VA. USA 22106-6148* Vol. XVI - No.5

Calendar

Meetings are normally held on the first Monday of each month at the Red Cross Building in Merrifield Va.

Monday	Oct 2	Meeting
Fri-Sun	Oct 27-29	OD 150 Horse Race
Monday	Nov 6	Meeting
Monday	Dec 4	Meeting
Monday	Jan 8	Meeting
Monday	Feb 5	Meeting

AMRAD publishes six Newsletter issues a year (on the first weekend of each odd-numbered month). Authors are encouraged to submit copy to the Editor by the first weekend of even months for the next Newsletter. The originals of figures, pictures, or graphics should be directed to the Publisher (Bill Pala) with a copy to the Editor (Deborah Borden).

The President's Ramblings

By André Kesteloot N4ICK

Trained observers will have noticed that digital signal processing (DSP) is the "in" subject these days. Now if you are one of the official AMRAD DSP gurus, such as David Borden K8MMO, Maitland Bottoms AA4HS, John Teller N4NUN, or Terry Fox WB4JFI, you need read no further. If you aren't listed above, you may just have some nagging questions, such as: "Is DSP good for me? If so, what do I do with it? How do I get some limited experience with this new technology, so that I may get a feel for its capabilities and decide whether to invest and/or investigate further?"

Some possible ham radio applications of DSP have been described in a very informative article published some time ago by Matjaz Vidmar YT3MV in VHF Communications (1).

In DSP, we want to use the digital computer to process (i.e., multiply, amplify, filter, store, etc.) an external-world event which usually presents itself in the analog mode (a sound, for instance). The mathematician Fourier (1768-1830) produced a set of transforms allowing us to translate time-domain representations into the frequency domain, and vice versa. These are called Fourier transforms and in the professional world out there, people nowadays make extensive use of Fast Fourier Transforms (FFTs). For more information on FFTs, read the other articles in this and previous issues of the AMRAD Newsletter.

To handle the outside world, we thus need an interface between the (analog) world and the (digital) computer. This analog-to-digital interface is often called an A-to-D converter.

In a recent AMRAD Newsletter (2) I described a simple, one- chip 10-bit A/D converter, housed in a little box which connected to the IBM XT through the printer port. Its main disadvantage was that it transmitted the information to the computer in serial form, and was thus inherently *slow*. (The printer port itself is slow.) In fact, Paul Rinaldo W4RI recently remarked to me that my simple board represented a major contribution to the little- known field of Slow Fourier Transforms.

This time, I shall describe a faster A/D converter for the IBM-XT clone. It features the MAX164, a CMOS chip (sporting built-in track-and-hold, on-chip oscillator circuitry, and voltage reference) recently introduced by Maxim Integrated Products. In the interest of speed, I installed it on a plug-in board and connected it directly to the IBM bus. Most of what you will need is included inside the chip, and the rest of the "glue" is available on the JDR PR-2 plug-in prototype board. (The latter has all the decoding circuitry already printed on it. It uses a 74LS00, a 74LS08, a 74LS85, a 74LS138, two 74LS244 and one 74LS245.) Note that the MAX164 accepts analog signals between -5 and +5 volts. With a maximum sampling rate of 100 Kilosamples per second, this chip is ideal for many audio applications.

Essentially, one could get the data from a given A/D chip in two ways, either (a) serial, which would entail a 12 bit string of data, plus overhead for each reading, or (b) parallel, where the whole data is put out on a bus at one time. The MAX164 is a parallel chip, and although there are four ways to operate it, I chose as being most appropriate for the IBM XT what Maxim calls the "ROM mode, 2 byte read."

Now the IBM XT has an 8-bit data bus, and our MAX164 is a 12 bit device. It is thus necessary to slice the 12 bits of data into two distinct bytes, one containing 8 bits, the other the remaining 4 bits plus 4 leading zeroes. "2 byte read" means that the chip outputs the data as two bytes of 8 and 4 bits respectively, instead of one 12 bit byte.

There are two ways to get 8-bit data from the chip (which are not really explained in the Maxim manual). In the "Slow Memory Mode," the IBM is put in a Wait state while the chip samples the input, whereas in the "ROM mode," the sampling is accomplished between reads. The Maxim chip doesn't really know which mode is being used. The only difference is that in the Slow Memory Mode, the BUSY line on the Maxim chip is connected to the I/O CH READY line on

the bus (to create a Wait state), whereas in the ROM Mode the Busy line is left unconnected. The IBM Technical Reference manual warns not to keep the 8088 in a Wait state for longer than 2 microseconds. Since the Maxim chip would keep it down for 8 microseconds, the ROM mode is preferred.

Hardware: Assuming you have a JDR board (or equivalent) with the decoding chips, you will need to wire the remaining unused gate of the 74LS08 as shown on the figure (pins 8,9 and 10) along with the MAX164 and a handful of components. The MAX164 comes in a 24-pin narrow DIP. If you cannot find one, simply solder a 16-pin socket at the end of a 8-pin one. Note that I used the IBM +5 volt and -12 volt lines for my prototype, and that this works very well for large (1 volt or better) input signals. For more minute signals, however, the digital noise present on the +5v and -12v becomes objectionable, and I recommend that you create a separate +5 volt line, using a 78L05 tied to the +12 volt IBM bus, and a small -12 volt power inverter using a 555 free-running oscillator operated from the +5 volt IBM bus.

This is how the board operates: referring to figure 1, when you read I/O location 300h, pin 15 of the 74LS138 goes low, bringing low pin 21 (-CS or Chip Select) and pin 19 (HBEN = High Byte Enable) of the MAX164. At the same time, the IBM 8088 microprocessor brings low the -I/O Read line, pin 20 of the MAX164 (I/O RD is available on pin B14 of the card connector. This signal, buffered, appears at pin 16 of the 74LS244). When these three pins (pins 19, 20 and 21) are low, a conversion cycle is initiated and the first byte is sent to the bus via the 74LS245. Then the software (see later) reads I/O location 304h. Pin 14 of the LS138 now goes low, and -CS goes low. -RD (pin 20 of the Max164) is low, as we are still in a reading cycle, but because HBEN (pin 19) is now high, a new conversion does not take place. Instead, the other byte is sent to the data bus.

The complete data is now available in memory for computation purposes.

Software: Every port on the IBM bus has a specific function (e.g., 3BCh is printer data out). Ports 300h to 31Fh are reserved for experimenters. My hardware is set up so that reading port 300h will return the least significant byte (and will start a sample), and 304h will return the most significant 4 bits. In Slow Memory mode, the LSB needs to be read first, then the MSB:

```
MOV DX,300h
IN AL,DX
MOV AH,AL
MOV DX,304h
IN AL,DX
XCHG AL,AH
```

In ROM mode, they need to be read backwards:

```
MOV DX,304h
IN AL,DX
MOV AH,AL
MOV DX,300h
IN AL,DX
```

Note that in the ROM mode, the first sample is old and needs to be discarded. The bytes are read backwards because reading 300h will start a sample, but the data won't be ready right away (which is why the first sample is bad). The next read will retrieve the data whose sample was initiated by the previous read. Make sure that you wait at least 8 microseconds before doing the next read (no software delay was necessary on a 4.77 MHz PC XT).

In the Slow Memory mode, reading port 300h will put the 8088 on hold until sampling is finished, which means that the data is available right away. The manual suggests that the Slow Memory mode will give cleaner results. Although putting the 8088 in a Wait state for longer than 2 microseconds was supposed to do something utterly horrible to the computer, nothing happened when we tried it.On a 4.77 MHz PC XT, using a driver written in assembly language, we can get a sampling rate of 45,000 samples per second. It is likely that this is the highest speed attainable with a plain-vanilla XT. Using the same program on a Turbo board running with a 8 MHz clock would probably bring the sampling rate to 100,000 samples, thus fully utilizing the MAX164.

Several programs written by our software brigade are available for this "MAXBOARD" on the AMRAD BBS at (703) 734-1387. Download the file MAXIM.ZIP. (It contains all the help you will need to capture incoming signals, display them on the screen, and perform some computing tasks, such as creating a signal twice the frequency of the incoming one. See figure 2.)

Construction: I would recommend you use the JDR board mentioned above as, for $29, it will considerably simplify your life (3). You can obtain the MAX164 chip in small quantities from DANCON (4). All other parts are readily available. You will want to keep separated the digital and analog ground returns, and only connect them together at one point.

Possible improvements:

(a) Add a D-to-A chip on the board (accessed through port 308h) and connect it directly to the speaker of the IBM so we can play Led Zeppelin songs backwards and listen for satanic messages.

(b) Find a safe and healthy way of using the Slow Memory mode.

(c) Add a linear buffer stage between the analog input and pin 1 of the MAX164.

I am thankful to John Teller N4NUN and Lawrence Kesteloot N4NTL for their major contributions to this project.

Let me know how you are doing, and happy hamcomputing!

DE André N4ICK.

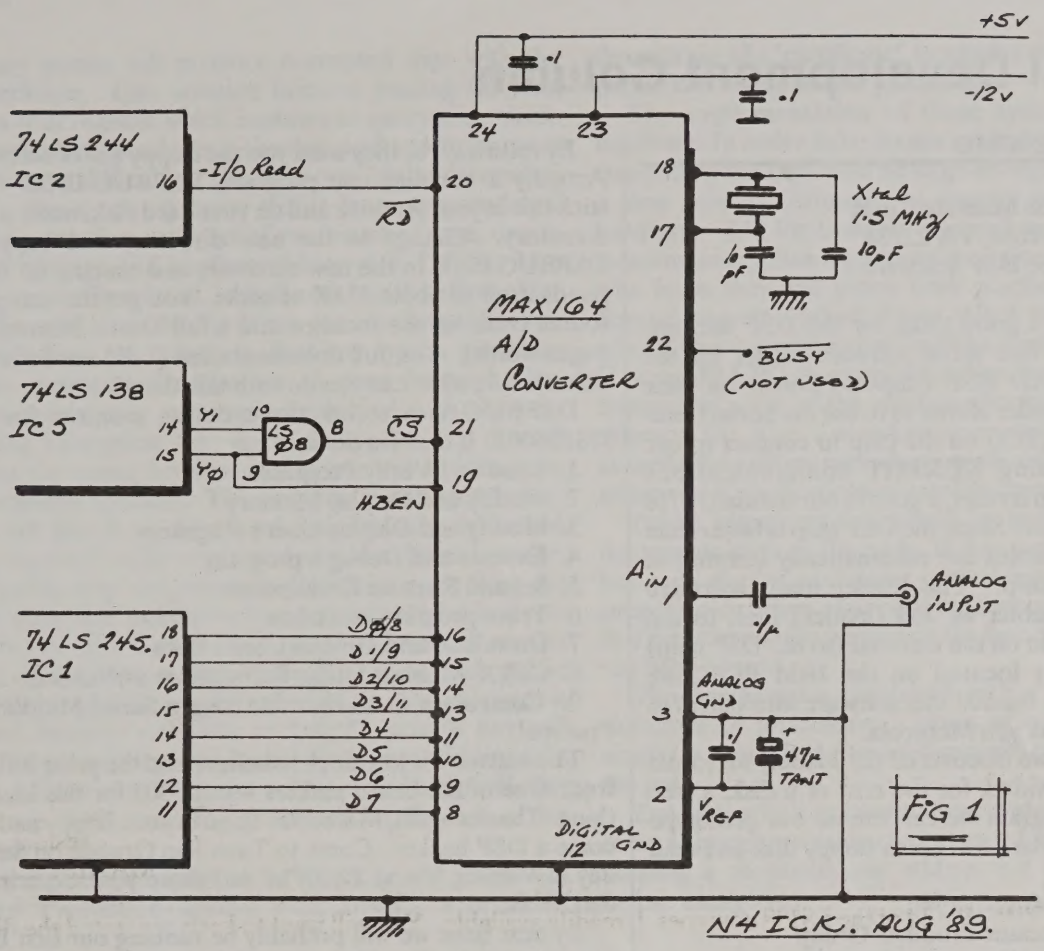
Notes:

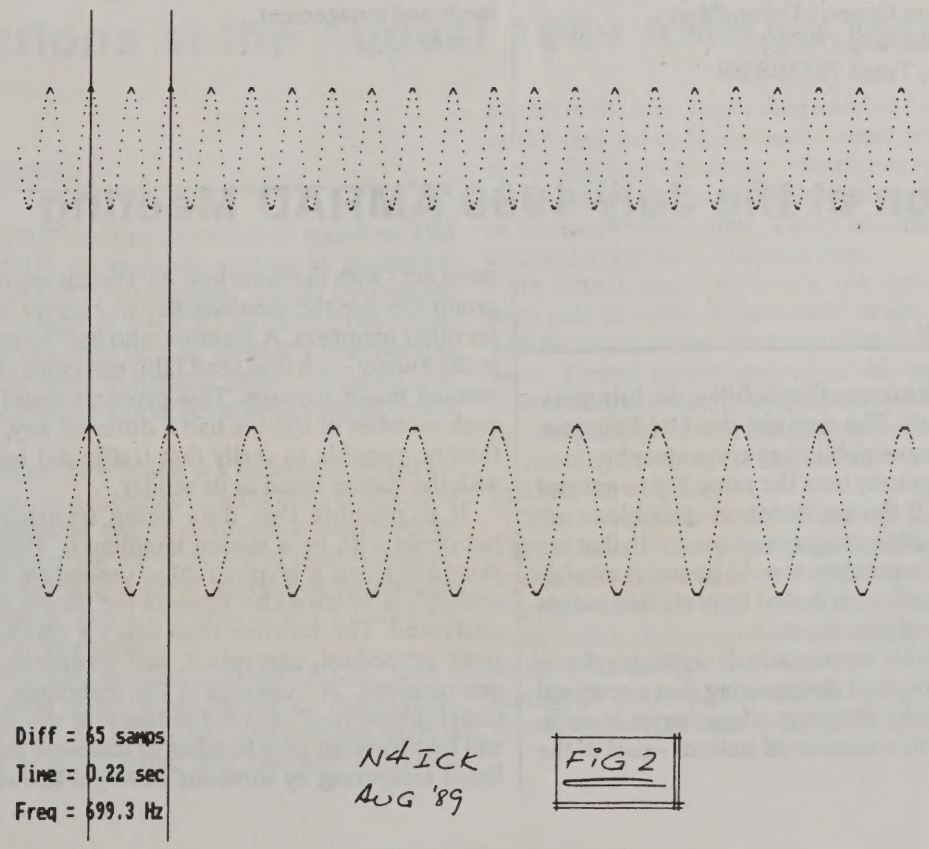
(1) VHF Communications 2/88. This is the English language version of UKW Berichte. It is available in the U.S. from Timekit, POBox 22277, Cleveland, Ohio 44122, telephone (216) 464- 3820.

(2) AMRAD Newsletter # XVI-No2, March 1989.

(3) JDR Microdevices, 2233 Branham Lane, San Jose, CA 95124. Telephone (800) 538-5000.

(4). DANCON Electronic Wholesalers, 304 Mill Street, Vienna, VA 22180. Telephone (703) 255-9755. (Ask to speak to Jeff Evans and refer to this article.)





N4ICK
AUG '89

FIG 2

The Digital Development Column

by David W. Borden, K8MMO

10808 Mantilla Court
Oakton, VA 22124

Free DSP Software

Motorola has done a good thing for the DSP hackers. First, they provided a free serial software loader for the DSP56001 (the AMRAD DSP Chip-of-Choice) on their Bulletin Board. This loader allows us to use the Serial Communications Interface (SCI) on the chip to connect to our IBM PC/Clone running KERMIT communications software. When the chip is reset, it goes to our onboard 2716 which contains the loader. Since the DSP chip is faster than lightning, fifteen wait states are automatically inserted at startup by the DSP chip. The loader loads software produced by the assembler as .lod (loader) files, to the desired memory location on the external (to the DSP chip) 45ns RAM (physically located on the IBM PC/Clone prototype board). We burned this software into our 2716 and plugged it in. Thank you Motorola.

Just about the time we discovered the loader, Motorola did it again. They provided, for the cost of a disk, a free DSPBUG Monitor program we can run on our prototype board. You take a standard 5 1/4 inch floppy disk and mail it to:

Motorola Inc.
Microprocessor Products Group
Digital Signal Processor Operation
6501 William Cannon Drive, West
Mail Stop OE314
Austin, Texas 78735-8598

By return UPS, they send you the floppy full of software. Actually it contains one program, DSPBUG.EXE. You stick this in your A: drive and on your hard disk, make a new directory. Change to the new directory and copy the DSPBUG.EXE to the new directory and execute it. It expands itself to about 515K of code. You get the complete source code for the monitor and a full User's Manual (78 pages worth). Yes, full documentation (I did not believe it either). Ok, what can you do with this thing?

DSPBUG is a stand alone debug monitor for the DSP56001. It enables the user to:

1. Load and Verify Programs
2. Modify and Display Memory
3. Modify and Display User's Registers
4. Execute and Debug a program
5. Set and Remove Breakpoints
6. Trace program execution
7. Download and Upload User's Data
8. Call 32 Monitor Utility Subroutines as Needed
9. Control up to 16 Boards Using a Serial Multiboard Protocol

This software is too much to believe and the price is right, free. One of the board makers wants $300 for this kind of thing. Thanks again, Motorola. Send in your floppy and become a DSP hacker. Come to Taco Rio Grande on Saturday in Vienna, VA at 12:30PM and share your experience and ideas.

By next issue we will probably be running our first DSP programs and will publish them right here for your amusement and amazement.

Presentation at the July 1989 AMRAD Meeting

A Synopsis

by Dick Barth, W3HWN

Because of the Independence Day holiday, the July meeting was held on the tenth. The speaker was Hal Feinstein, WB3KDU, whose topic was public-key cryptography.

Conventional cryptography uses the same key to encrypt and decrypt a message. If the key becomes available to unauthorized parties, then all messages encrypted with that key are compromised. Keys must therefore be guarded carefully and new ones periodically distributed by protected means (encrypted circuits, armed guards, etc.).

A major difficulty with conventional cryptography is authentication - the problem of determining that a received message was in fact sent by the party whose name is on it. Messages are often sent to a number of stations - part of the same net - with the same key. As a result any member of that group can use the common key to encrypt spurious traffic for other members. A member who has "turned" - gone over to the enemy - can thus send false messages at critical times, causing major damage. This problem could be avoided if each member of the net had a different key, since it would then be possible to verify that traffic did indeed originate with the station listed as its sender.

It is possible that data being transmitted could be tampered with by a station handling it. Protection can be provided using a cryptographic procedure called "cryptosealing", in which a checksum of the data is taken and then encrypted. The receiver then takes a checksum using the same procedure, encrypts it, and comparing that with the one received. A weakness of this technique, termed "functional degeneracy", lies in the fact that the same checksum will be produced by a number of different data sets. Intelligent tampering by someone having a knowledge of these

degeneracy points will produce corrupted data with the same checksum. One solution involves passing the data through a shift register which contains an encryption block. This complication produces a checksum whose degeneracy points are more difficult, or almost impossible, to determine.

Public key cryptosystems differ from conventional methods in that they use a set of two matched keys, one to encrypt a message and another to decrypt it. The two keys are mathematically related, in that each has the effect of inverting the work done by the other, but ideally neither can be derived from the other. A dual-key system produces automatic message authentication. A given decryption key will unscramble only messages scrambled with the corresponding encryption key; if the decryption key doesn't work then the wrong key was used to encrypt and the message is probably spurious. This property is particularly useful in a net situation, since each receiver can determine whether received traffic originated with the claimed sender.

The "public-key" cryptosystem derives its name from the fact that each user publishes, in a publicly available directory, only one of his set of two matched keys. A sender encrypts an outgoing message using the recipient's public key and then again using his private key. The message is protected because only the recipient's private key will remove the scrambling introduced by the use of his public key. It is authenticated because only the sender's public key will remove the scrambling generated by the use of his private key.

The system is vulnerable to an attack on the public-key directory. Protection against such attacks is provided through use of a "certificate", in which a trusted server issues public keys using its own private key.

The implementation of these systems is limited by hardware. In order to be secure against cracking by current computers, the keys must be large enough that encryption is a slow process. Attempts to encrypt by using dedicated hardware - LSI for example - haven't worked out well. Alternative encryption algorithms are being produced by those who hope they will prove both practical and profitable. Several haven't worked. Some failed rather spectacularly when it was shown that they could be cracked using an Apple II or an IBM-PC. A recent Canadian development appears immune to some of the solution shortcuts that caused the others to fail. Also, a new American product is being touted as useful for military spread spectrum usage as for as for such commercial applications as satellite command links.

Public key encryption would be useful to authenticate documents such as ID cards. If the card had an encrypted bar code display containing critical information about the proper holder, then a bar code reader could be used to verify that the information printed on the face of the document had not been altered.

The presentation continued with a discussion of how violations of procedures - reuse of encryption keys and similar boners - had damaged security of various nations in the past.

Data integrity is not normally provided by public-key systems, although provision can be made for it by externally calculating a checksum and adding that to the data stream before encryption.

Presentations at the August 1989 AMRAD Meeting

A Synopsis

by Dick Barth, W3HWN

The August 7, 1989 meeting featured as speakers Hal Feinstein (WB3KDU) on message routing in networks; David Borden (K8MMO) on Digital Signal Processing (DSP); and André Kesteloot (N4ICK) on spread spectrum.

Hal began by noting that he had invited a heavyweight routing guru from his office to speak at a future meeting, and was presenting this evening a brief tutorial on routing basics as an introduction.

Diversity code route combining is a technique which involves sending copies of a message (packet) to a destination via a number of different routes. At certain intermediate nodes multiple copies of the message will be received via alternate routes. Longitudinal error correction techniques can be used to provide error correction, and a corrected message can then be forwarded.

Virtual circuit routing, used in X.25 and AX.25, uses a "hard-wired" or fixed route from source to destination. Datagram techniques, on the other hand, route various datagrams (packets) along paths that can vary from one packet to the next, depending on what the network perceives as the shortest of fastest end-to-end path at any given moment. This latter technique clearly can result in packets that arrive at their destination out of sequence, get lost en route, or have errors on arrival. There are other techniques which combine these two in various ways.

In virtual circuit networks, the optimum path between each pair of nodes is calculated at one or more main sites and the information then distributed throughout the network. Tymnet uses a network of this type. Telenet, on the other hand, uses a two-level hierarchy. Dijkstra's algorithm for computing the shortest path is used in ARPANET.

Routing involves two separate functions: route calculation and forwarding.

There are three kinds of message common in networks. The so- called "nearest neighbor" message is used to inform nodes of what nodes are connected directly to them. Another message type provides information on the time delays (link distances) between nodes. Distribution of this information allows each node to know the interconnects among all other nodes, and to calculate for itself the optimum route from itself to anyplace else. For large networks,

the information table becomes extremely large and often unwieldy.

Alternatives to this approach include "Area routing", which provides each node with information regarding only those nodes within a limited distance, or within its horizon. Another is "landmark hierarchy" in which only certain nodes, called landmarks, contain complete routing information. Other routers know only how to route a packet to the nearest landmark.

As things change in a network, information about these changes has to be sent around the network so that the various routers can choose a new "optimum" path. The fact that information has a finite (non-zero) dissemination time leads on occasion to propagation anomalies, in which messages might be mis-sent or even end up in a loop.

Dave Borden next provided an update on the progress of AMRAD's DSP hackers. He noted that Motorola provides an excellent DSP reference manual (which is big enough to approximate a textbook) which can be gotten free by calling an 800 number.

The group needs good-quality A/D converters for use in their experiments. Dave recently sent away to Motorola for four ceramic chips (at $70 a copy) which arrived within a week. They provide a conversion rate of better than 100 KHz, 16 bits, and a 3-wire interface to the DSP prototype board.

A TI analog interface circuit costs $46, has a 14-bit A/D converter and a 14-bit D/A converter, and several filters. Two chips were obtained and are being tested with the DSP prototype board.

The group has also developed a loader program that will allow the upload of the DSP board and its associated memory. This code has been put into a 2732 ROM and work using it is in process. A free software monitor program was obtained from Motorola and it is being tested.

André presented a tutorial he had developed for presentation to groups with no experience in spread spectrum (SS). It summarized his work and that of other AMRAD members, as covered in a number of previous presentations at club meetings.

SS involves the radiation of a bandwidth which is much larger than required by traditional modulation schemes to convey the same information. Where a normal FM signal might require fifteen KHz to transmit a signal, the same information might occupy several MHz using SS. Most often, the SS bandwidth is independent of the information signal. Synchronized reception is often required in order to decode the transmitted information. Ultra-wide frequency modulation is often excluded from the definition of SS because it fails to meet this third test.

The two major types of spread spectrum are frequency hopping and direct sequence. Hopping involves the periodic change of a transmitter's center frequency in a predictable but pseudorandom manner. The instantaneous bandwidth is the same as with traditional modulation methods. The center frequency changes rapidly (from a few to hundreds of times a second) over a set of perhaps dozens of frequencies, so the effective bandwidth is much larger. The hopping procedure reduces the effect of a single narrow-band interferer because it is seen only part of the time, with the hit ratio being dependent on the hopping sequence and the size of the hop suite. Deliberate jamming also becomes more difficult because if the hop rate is high enough and the jammer sufficiently distant, the transmitter will have moved off a frequency before the jammer first hears him on it. Following jammers become useless, and only broadband or intelligent jamming techniques will work. Since deliberate jamming is not a regular occurrence on the amateur bands, the usefulness of frequency hopping SS to amateurs remains questionable.

Direct sequence (DS) techniques involve the use of two modulating signals. The information signal is used in the normal manner, but a second pseudorandom modulating signal with a very high data ("chip") rate causes spreading. This pseudorandom sequence must be duplicated at the receiver in order to provide de-spreading and detection. This requires a duplicate sequence generator which is synchronized to that at the transmitter.

Generation of the DS signal usually uses (in amateur designs) a balanced modulator made up of ring-connected diodes and driven both by the pseudorandom square wave and the information signal. The output spectrum using only the spreading modulation is a $[(\sin x)/x]^2$ curve whose precise characteristics depend on the pattern of the modulating signal.

In order to receive a DS signal, the receiver requires a copy of the spreading signal. That is, it must have a duplicate of the pseudorandom sequence used to spread the signal, and that duplicate must be synchronized with that at the transmitter. Amateur SS is constrained to limited geographic areas due to FCC rules limiting its use to the UHF frequencies. Hence, it was originally thought that synchronization could be best derived through the use of a local broadcast outlet as a reference. An AM station was tested in this application but the results were found wanting. The final design involved a design in which the receiver generated its own binary sequence and synchronization was derived from the received signal itself. The method by which this was done has been described in previous editions of the AMRAD newsletter, in QEX and in QST and will not be repeated here.

There followed a demonstration of the synchronized oscillator used to lock the receiver to the transmitter frequency. Again, see previous editions of the newsletter for a description of the circuit.

The test equipment used by André worked several miles using about half a watt on 440 MHz. Range of this type of equipment will be expanded considerably when a 900 Mhz spread spectrum repeater, currently being built by Chuck Phillips (N4EZV), is in operation. André and Chuck also have plans to experiment with SS television, and mobile SS operation is also planned.

Old Dominion Ride

by Dave Rogers N4JGQ

The next Old Dominion event is the 150/100/50-mile ride on October 27-29. This is a three day (day light) event of about 11 hours in duration each day. 50 miles are run each day with the 150-mile event starting on Friday the 27, the 100-mile event starts on Saturday, and the 50-mile event starts and ends on Sunday. The course is within the Fort Valley area, southwest of Front Royal, Va. This event is the most fun for the least investment of time of any of the rides that Old Dominion puts on. Volunteers are needed for each day to staff Net Control and way points. All points can be reached by vehicles. The Old Dominion will provide lunches for all workers. If you would like to help, please give me a call at 703-237-5465.

Dancon Electronic Wholesalers, Inc.

304 *Mill Street*
Vienna, Virginia 22180

Jeff Evans *(703) 255-9755*

Your own
business card
could appear here
in our next issue.
Contact us for our
r a t e s.

AMRAD MEMBERSHIP APPLICATION/RENEWAL

Mail to: Membership, AMRAD, P.O. BOX 6148, McLean, Va. 22106-6148

ANNUAL DUES: Regular $15; 2nd in family at same address add $8; Canada and Mexico add $2; foreign surface add $2.30; foreign air mail add $8; currency is US dollars. A donation to the WD4IWG/R repeater fund is appreciated if you use the repeater. Please make checks payable to AMRAD.

Name_____ Amateur Call_____ License Class_____

Street Address_____ App#_____

City_____ State/Prov_____ ZIP/PC_____

Country_____ Home Phone (___)____-_____ Work Phone (___)____-_____

Special Interest Groups: please indicate your interest.

__Deaf Communications __RTTY __Spread Spectrum __Packet Radio __Digital Signal Processing

Network: CompuServe ID:_____

_____ ID:_____

Packet Radio mail box_____ ARRL member?____

AMRAD

The Amateur Radio Research and Development Corporation (AMRAD) is a worldwide club of several hundred amateur radio and computer experimenters. It is incorporated in Virginia and is recognized by the U.S. Internal Revenue Service as a tax-exempt scientific and educational organization.

AMRAD's Purpose

The purpose of AMRAD is to: develop skills and knowledge in radio and electronic technology; advocate design of experimental equipment and techniques; promote basic and applied research; organize technical forums and symposiums; collect and disseminate technical information; and provide experimental repeaters.

Meetings

Meetings are on the first Monday of each month at 7:30 P.M. at the Red Cross Building, Merrifield, VA. If the first Monday is a holiday, an alternate date will be announced in the newsletter. Except for the annual meeting in December, meetings are reserved for technical talks - not business.

WD4IWG Repeater

WD4IWG/R is an open repeater for FM voice and digital communications, especially for experimental modes. It is located in McLean, VA. It features an autopatch available to licensed members. Frequencies are: 147.81 MHz in and 147.21 MHz out. The repeater director is Jeff Brennan, WB4WLW.

Westlink Report

The Westlink report is aired every Sunday night at 8:00PM on the WD4IWG repeater.

WB4JFI-5 Digipeater

WB4JFI-5 is a 1200-baud AX.25 Level 2 digipeater operating on 145.01 MHz. It is located in northwest Washington DC at Wisconsin Ave. and River Rd. The digipeater is mounted at an elevation of 350 feet on the south leg of the WUSA/WJLA TV tower.

Data Address

Our CompuServe/Micronet number is [72345,1050].

AMRAD Computer Bulletin Board System

AMRAD CBBS, (703) 734-1387, is operated by Lawrence Kesteloot, N4NTL. The system accepts calls at 300, 1200 and 2400 baud. The data path settings are 8 data bits, 1 stop bits, and no parity.

HEX Bulletin Board System

Handicapped Education Exchange is operated by Dick Barth, W3HWN. HEX accepts TDD/300 baud at (301) 593-7033,and 300/1200 baud at (301) 593-7357. ASCII is 8N1

Affiliations

AMRAD is affiliated with the American Radio Relay League (ARRL), Foundation for Amateur Radio (FAR), Northern Virginia Radio Council (NOVARC) and the Mid-Atlantic Repeater Council (T-MARC).

AMRAD Newsletter

The AMRAD newsletter is mailed six times a year to members and other clubs on an exchange basis. Technical articles, product announcements, news items, and other copy relating to amateur radio and computing is welcome. Honorariums of one year free membership are given for original material accepted. Maximum of one year per year. Classified ads are free to members. Commercial ad inquires are invited. The editor reserves the right to reject or edit any portions of the copy. Items should be mailed to Editor, AMRAD Newsletter, P.O. Drawer 6148, McLean, VA 22106. Full permission for reprinting or quoting items is granted provided that credit is given to both the author and the newsletter. Membership in AMRAD is $15 annually ($8 for second member of same family). Mailing to U.S. and possessions is by 3rd Class bulk mail. Canadian and Mexican addresses add $2 for postage. Overseas readers add $8 for air mail or $2.30 for surface.

AMRAD Officers for 1989 are:

André Kesteloot	N4ICK	President, Director	(703) 356-5519
Terry L. Fox	WB4JFI	Vice-President,Director	(703) 698-7621
Gerald Adkins	N4GA	Treasurer	(703) 538-6936
William P. Pala Jr.	WB4NFB	Secretary, Director	(703) 764-3710
David W. Borden	K8MMO	Director	(301) 859-0400
Jeffrey Brennan	WB4WLW	Alternate Director	(703) 698-7924
Elton A. Sanders	WB5MMB	Director	(703) 281-1298
John Teller	N4NUN	Alternate Director	(703) 938-5369

Amateur Radio Research and
Development Corporation
P.O. Drawer 6148
McLean, VA USA 22106-6148

Nonprofit Organization
U.S. Postage
Paid
McLean,, Virginia 22101
Permit No. 1511

9/90 - 645 -